Xerox Docket No. R/97005Q

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE HONORABLE BOARD OF PATENT APPEALS AND INTERFERENCES

In re the Application of

Jean-Marc ANDREOLI et al.

Application No.: 09/421,846

Filed: October 20, 1999

On Appeal from Group: 2176

Examiner: W. Bashore

Docket No.: 109619

For: DOCUMENT CONSTRAINT DESCRIPTORS OBTAINED FROM USER SIGNALS INDICATING ATTRIBUTE-VALUE RELATIONS

APPEAL BRIEF TRANSMITTAL

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Attached hereto are three (3) copies of our Brief on Appeal in the above-identified application.

The Commissioner is hereby authorized to charge Deposit Account No. 24-0037 in the amount of Three Hundred Thirty Dollars ($330.00) in payment of the Brief fee under 37 C.F.R. 1.17(f). In the event of any underpayment or overpayment, please debit or credit our Deposit Account No. 24-0037 as needed in order to effect proper filing of this Brief.

For the convenience of the Finance Division, two additional copies of this transmittal letter are attached.

Respectfully submitted,

James A. Oliff
Registration No. 27,075

Klifton L. Kime
Registration No. 42,733

JAO:KLK/sxb
Date: February 2, 2004

OLIFF & BERRIDGE, PLC
P.O. Box 19928
Alexandria, Virginia 22320
Telephone: (703) 836-6400

DEPOSIT ACCOUNT USE AUTHORIZATION
Please grant any extension necessary for entry;
Charge any fee due to our Deposit Account No. 24-0037

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE HONORABLE BOARD OF PATENT APPEALS AND INTERFERENCES

In re the Application of:

Jean-Marc ANDREOLI et al.

Application No.: 09/421,846

Filed: October 20, 1999 Docket No.: 109619

For: DOCUMENT CONSTRAINT DESCRIPTORS OBTAINED FROM USER SIGNALS INDICATING ATTRIBUTE-VALUE RELATIONS

BRIEF ON APPEAL

Appeal from Group 2176

OLIFF & BERRIDGE, PLC
P.O. Box 19928
Alexandria, Virginia 22320
Telephone: (703) 836-6400
Attorneys for Appellants

TABLE OF CONTENTS

I. INTRODUCTION

This is an Appeal from a final Office Action mailed July 29, 2003 finally rejecting claims 1-16. No claims are allowed.

A. Real Party In Interest

The real party in interest for this Appeal in the present application is Xerox Corporation, by way of an Assignment recorded at Reel 010643, Frame 0514.

B. Statement of Related Appeals and Interferences

There are presently no appeals or interferences, known to Appellant, Appellant's representative or the Assignee, which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

C. Status of Claims

Claims 1-16 are pending. Claims 1-16 stand finally rejected and are on appeal.

D. Status of Amendments

The Amendment filed on July 31, 2002, which was entered, amended the specification and amended claims 1 and 13. The Amendment filed on December 24, 2002, which was entered, further amended claims 1 and 13. No other Amendments have been filed.

II. SUMMARY OF THE REJECTION AND APPLIED REFERENCES

A. The Invention

The present invention relates to techniques that obtain constraints to facilitate searching, retrieval, storage and conversion of electronic documents on heterogeneous document management systems.

A typical user of electronic information search and retrieval systems has difficulty formulating sorts (a unary relation expressing a property of a single entity) and features (a binary relation expressing a property relating two entities) that will produce a desired document constraint.

This invention alleviates constraint production problems by providing techniques that obtain document constraint descriptors for documents from user signals.

A document constraint descriptor includes information about a set of one or more constraints that documents could satisfy. Instead of requiring the user to provide a set of sorts and features, the techniques allow the user to provide attribute-value relations, which are relatively easy for typical users to provide. The techniques then convert the attribute-value relations to logical relations such as sorts and features from which a constraint descriptor can be obtained.

The new techniques are implemented in a method for obtaining document constraint descriptors from user signals. The method receives user signals indicating a set of attribute-value relations that apply to documents. The method can uses the user signals to obtain logical relations equivalent to the attribute-value relations. The method then uses the logical relations to obtain a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations.

The document constraint descriptor is stored in memory and the method presents an image that includes an item representing the descriptor, receives a user signal indicating the item, and, in response, obtains the stored descriptor.

The method solves the set of constraints to obtain a solution and uses the solution to obtain document references indicating electronic documents in a repository accessible through a network. The method presents an image that includes items representing the document references and, in response to a user signal indicating one of the items, retrieves the item's electronic document. A portion of the electronic document is displayed and/or the electronic document is printed.

The new techniques may also be implemented in a machine. In general, such a machine includes a processor and user interface circuitry for providing user signals to the

processor. The processor operates as described above, receiving user signals indicating a set of attribute-value relations applicable to documents, using the user signals to obtain logical relations, and using the logical relations to obtain a document constraint descriptor.

The new techniques provide convenient ways for ordinary users to produce document constraint descriptors. The descriptors may be used to specify search requests, answers to requests, and the state of retrieval agents.

The following definitions are helpful in understanding the broad scope of the invention, and the terms defined below have the indicated meanings throughout the application, including the claims.

A "processor" or "processing circuitry" is a component of circuitry that responds to input signals by performing processing operations on data and by providing output signals. A processor may include one or more central processing units or other processing components. A processor can be a general purpose processor or a special purpose processor.

A "user interface" or "user interface circuitry" is circuitry that can provide signals from a user.

The notions of "constraint" and "satisfy" are related: A constraint is a condition that, when met, is satisfied. A "constraint that documents can satisfy" is therefore a condition that could be met by a document. A constraint can be a logical combination of constraints, such as a conjunction of a set of subconstraints, in which case the constraint "includes" the subconstraints. For example, constraints that documents can satisfy may be expressed as logical combinations of simpler constraints such as attribute-value relations, where each attribute-value relation is between an attribute that a document could have and a set of at least one value of the attribute. A constraint is "inconsistent" if the constraint cannot be met because of its logical structure; if inconsistency of a constraint can be determined from logical structure, it is unnecessary to search or check whether a document can be found that

meets the constraint - no document could possibly meet the constraint. A constraint that is not inconsistent is "satisfiable" even though the constraint may not in fact be satisfied by any stored document.

A "constraint descriptor" is an item of data that defines a constraint. A "document constraint descriptor" is a constraint descriptor defining a constraint that is applicable to documents.

An operation "compiles" a constraint if the operation operates on one or more items of data that provide information about a constraint to obtain a constraint descriptor that defines the constraint.

The terms "attribute" and "value" are related: An "attribute" is a characteristic that may have a "value". A "document attribute" is an attribute that documents could have. A "set of values" is any combination of one or, more values. For example, a set could be a single value, a range of values, or a set of two or more non-contiguous values.

An "attribute-value relation" is an association between an attribute and a set of values the attribute could have. If the attribute is a document attribute, the attribute-value relation could apply to documents.

A "logical relation" is a relation between elements, where the relation can be evaluated as true or false. Sorts and features are examples of logical relations.

A set of logical relations is "equivalent" to a set of attribute-value relations if the logical relations are evaluated as true only if the attribute-value relations are met and are evaluated as false only if the attribute-value relations are not met.

Similarly, a set of constraints is equivalent to a set of logical relations only if the constraints are only satisfied when the logical relations are evaluated as true and the constraints are only not satisfied when the logical relations are evaluated as false.

A "solution" of a constraint or a set of constraints is an item of data that indicates whether the constraint or set of constraints is inconsistent or satisfiable and, if satisfiable, indicates a less redundant version that is equivalent to the constraint or set of constraints. In this context, the solution is "equivalent" to the constraint or set of constraints if the solution can only be satisfied if the constraint or set of constraints is satisfied and vice versa.

An operation "solves" a constraint or a set of constraints if the operation obtains a solution of the constraint or set of constraints.

A great deal of research in knowledge representation has been focused on identifying fragments of classical logic in which satisfiability is algorithmically decidable. The trade-off here is between expressive power and tractability: The empty fragment, for example, is obviously tractable, but it is not very expressive.

The most popular fragment which emerged is known as "feature constraints". The satisfiability problem in this case is also known as "feature constraint solving".

As is known, feature constraints can be built from atomic constraints that are either sorts or features. A sort is a unary relation, expressing a property of a single entity. For example, P : person expresses that an entity P is of sort person. A feature is a binary relation expressing a property linking two entities. For example, P : employer $\rightarrow$ E expresses that entity P has an employer, which is an entity E . Apart from sorts and features, most feature constraint systems also allow built-in relations such as equality and inequality, and such relations are also referred to herein as "built-in predicates" or "built-in constraints".

The full fragment of feature constraints, where the atomic components mentioned above are allowed to be combined by all the logical connectives (conjunction, disjunction, negation and quantifiers), although very expressive, is hardly tractable. A subfragment called "basic feature constraints" (BFC) has been considered, where negation and disjunction are simply forbidden. Efficient constraint solving algorithms have been proposed for this

sub-fragment. However, a drawback is that the complete absence of negation puts strong limitations on the kind of operations a knowledge broker may wish to perform.

A signed feature constraint is composed of a positive part and a list of negative parts, both of them being basic feature constraints.

The processor of a device can obtain a document constraint descriptor from user input signals in a number of ways.

Fig. 3 of the application shows an image presented by display circuitry of a fixed computing device while a user is entering a query, e.g., "books or articles after 1990 in which the title contains 'constraints' but does not contain Internet".

Each element of the query is added to the current specification of the query, and the image in Fig. 3 also includes a box 37 that contains the current specification. The image also includes a button 38, which the user can select to launch a search based on the current specification of the query.

Fig. 4 of the application shows operations that can be performed by the processor of a device that presents images as in Fig. 3 in obtaining a document constraint descriptor. As shown in Fig. 3, the query elements can be attribute-value relations such as "title contains 'constraints'". Several other types of attribute-value relations are described in the application in relation to Fig. 5.

When the user launches a search by selecting the button 38, thus confirming the current specification, the processor converts each query element in the existing list to a logical relation. This can be accomplished by producing sorts and features as described above. For example, a data structure could be stored containing predefined mappings from standard query elements or standard types of query elements to logical relations. The processor can also store data indicating the logical relations in memory.

Once the logical relations are obtained, the processor can automatically compile a signed feature constraint from the logical relations. This can be thought of as beginning to solve a constraint that is equivalent to the relations. The compiled feature constraint can thus be an item of data that includes signs occurring in the stored relations, and is therefore a type of document constraint descriptor. The processor stores the compiled feature constraint in memory.

User signals indicating attribute-value relations could be provided interactively in various other ways. For example, the user could provide signals to the processor of a portable computing device, using a keyboard or a touchscreen user interface on which lists of items are displayed and can be navigated or selected using scrolling and control buttons.

Fig. 5 of the application shows query sheet 50 that a user can mark to indicate a query.

In a field 51, the user can mark boxes, such as boxes 58, to indicate the type of a document. In the illustrated example, the user has indicated that the value for the document's type should be any type other than "journal".

Then the processor can convert each query element in the existing list to a logical relation. This can be accomplished in the same manner described above.

Once the logical relations are obtained, the processor can automatically compile a signed feature constraint from the logical relations.

Fig. 7 of the application illustrates operations performed by the processor of a computing device in using a document constraint descriptor, such as a feature constraint, to retrieve document references and in displaying or printing documents.

In box s91 of Fig. 7, the processor receives a feature constraint from a device, such as from a device that obtained the feature constraint in accordance with Fig. 4 or Fig. 6 of the

application. The processor can receive the feature constraint in a data packet from a portable computing device or in any other appropriate way.

In box s92 of Fig. 7, the processor receives further user signals requesting a search for documents satisfying the feature constraint. The user signals can again be received in any appropriate way, such as by presenting an image that includes an item representing the feature constraint and receiving a user signal selecting the item.

In response, the processor can solve the feature constraint using the techniques described above for solving basic feature constraints and signed feature constraints. Therefore, solving the constraint in box s92 can be thought of as completing the solution process that was begun by compiling in box s45 or box s65 of Figs. 4 and 6, respectively.

If the processor obtains a solution, the solution can be used to formulate a search request, which the processor can then provide in a call to search engine routines it also executes.

A "query" is a request for information in some form entered by a user to a database system. A noted above, Figure 3 of the application shows an exemplary user query, e.g., "books or articles after 1990 in which the title contains "constraints" but does not contain "internet." As pointed out on page 28 of the specification, "title contains 'constraints'" is an attribute value relation, indicating that the document attribute "title" has a value that includes the wore "constraints." The query element "date after 90" also is an attribute value relation, indicating that the document attribute "date" has a year value greater than 90. The Specification describes several other types of attribute-value relations in relation to Fig. 5. For example, page 32, first paragraph, discloses the situation where a user has indicated that the document attribute "type" has a "type" value other than "journal."

According to the invention, a set of logical relations is generated that is equivalent to user query generated attribute-value relations and document descriptor constraints are

generated that are equivalent to the generated set of logical relations. Examples of this are disclosed on pages 21 and 22 of the application. The steps involved are shown, for example, in Fig. 4. In particular, each query element is converted to a logical relation. Then, a signed feature constraint is compiled from the set of logical relations generated.

B. The Claims

Claim 1 recites a method for obtaining document constraint descriptors based on user signals, comprising: (A) receiving user signals indicating a set of attribute-value relations that can apply to documents; (B) using the user signals to obtain, without requiring user intervention, logical relations equivalent to the attribute value relations, the logical relations comprising at least one of a sort and a feature; and (C) using the logical relations to obtain, without requiring user intervention, a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations.

Claim 13 recites a machine for obtaining document constraint descriptors based on user signals, the machine comprising: (a) a processor; and (b) user interface circuitry for providing user signals to the processor, the processor operating to: (1) receive user signals through the user interface circuitry indicating a set of attribute-value relations that can apply to documents; (2) use the user signals to obtain, without requiring user intervention, logical relations equivalent to the attribute-value relations, the logical relations comprising at least one of a sort and a feature; and (3) use the logical relations to obtain, without requiring user intervention, a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations.

C. The Rejections

The final Office Action rejects claims 1-5, 8 and 13 under 35 USC §103(a) as unpatentable over U.S. Patent 5,794,233 to Rubinstein (hereinafter, "Rubinstein '233") in view of U.S. Patent 5,693,938 to Wilson et al. (hereinafter, "Wilson").

The final Office Action rejects claims 6 and 7 under 35 USC 103(a) over Rubinstein '233 in view of Wilson as applied against claim 1, and further in view of U.S. Patent 5,404,294 to Karnik (hereinafter, "Karnik").

The final Office Action rejects claims 9-12 under 35 USC 103(a) over Rubinstein '233 in view of Wilson as applied against claim 1, and further in view of U.S. Patent 5,721,897 to Rubinstein (hereinafter, "Rubinstein '897").

For the reasons detailed below, it is respectfully submitted these rejections are improper and should be reversed.

D. U.S. Patent 5,794,233 to Rubinstein

Rubinstein '233 discloses a method and apparatus to identifying one of a plurality of documents by permitting a computer user to browse documents by prompted keyword phrases. Once a query expression is selected by a user, e.g., from an automatically generated list of keyword phrases in the plurality of documents, the query selection is used to identify one of the plurality of documents and an abstract of the identified document is presented to the user. Rubinstein uses a commercially available linguistic analysis tool to linguistically analyze documents - see column 6, lines 34+.

E. U.S. Patent 5,963,938 to Wilson et al.

Wilson discloses an apparatus and method to select criteria for use in execution of a logical function such as a search, filter, sifting, or the like, using a graphical user interface in which logical operators and Boolean operators are presented in distinct graphical locations demonstrating to an unsophisticated user the distinction between Boolean relations and logical relations.

A dialogue box is provided that may include argument windows related by a logical operator button. Two corresponding arguments related by a corresponding logical operator form a logical relation. Argument windows and operator buttons may each provide a menu

when selected, thus presenting options for a user to select from. A Boolean operator may relate two logical relations in a Boolean relation.

In column 10, lines 60+, Boolean operators are defined as a Boolean AND and a Boolean OR, and the other operators, including equality, inequality, and comparatives such as greater than, exclusive less than, inclusive greater than, inclusive less than, and the like, are disclosed as logical operators. Groups of Boolean relations may be related by other Booleans in inter-group Boolean relations.

A language text equivalent (LTE) of the combination of all relations formed by a user may be presented in a language or text window within a dialogue box. The LTE provides feedback to a user to tell a user whether the relations produced are actually what the user seeks or needs. A user may further edit the relations.

F. U.S. Patent 5,404,294 to Karnik

Karnik extracts information stored in a database and placed at user specified positions on a pre-printed form or form-type document and to extract information from a pre-filled form or a form-type document using user-created Tags, where a Tag is defined as a record which contains a specific set of operations and functions stored as fields.

G. U.S. Patent 5,721,897 to Rubinstein

Rubinstein '897 discloses an apparatus and method for identifying one of a plurality of documents stored in a computer-readable medium. The method includes the steps of automatically identifying for a user key word phrases in the plurality of documents, prompting the user to construct a query expression in which at least one of the keyword phrases is an operand, and identifying one of the plurality of documents based on the query expression.

The identified document is presented to the user in the form of an abstract. Identification of the keyword phrases and generation of the abstract are accomplished by

linguistically analyzing the plurality of documents. An improved user interface provides the capability to display either or both key words and key phrases on the display screen in separately scrollable display areas. These separately scrollable display areas are dynamically sized to render visible the selected text.

A set of dynamically created tabs in a tabbed index provide a means to index into the content of each display area. The font of the selected and displayed text is dynamically set to maximize the display area. The plurality of documents from which key words or key phrases are taken may be pages from the World Wide Web. A concept editor allows key words or key phrases to be grouped under a concept identifier.

III. THE ISSUES ON APPEAL

1. Whether, under 35 U.S.C §103(a), claims 1-5, 8 and 13-16 would have been obvious over Rubinstein '233 in view of Wilson.

2. Whether, under 35 USC §103(a), claims 6 and 7 would have been obvious over Rubinstein '233 in view of Wilson and further in view of Karnik.

3. Whether, under 35 USC §103(a), claims 9-12 would have been obvious over Rubinstein '233 in view of Wilson and further in view of Rubinstein '897.

IV. GROUPING THE CLAIMS ON APPEAL

All claims stand or fall together.

V. ARGUMENT

A. The Law (35 USC §103(a) (Obviousness))

The Supreme Court in Graham v. John Deere, 383 U.S. 1 at 18, 148 USPQ 459 at 467 (1966), set forth the basic test for patentability under 35 U.S.C. §103:

> Under §103, the scope and content of the prior art are to be determined; differences between the prior art and the claims at issue are to be ascertained; and the level of ordinary skill in the pertinent art resolved. Against this background, the obviousness or nonobviousness of the subject matter is determined. Such secondary considerations as commercial success, long felt but unresolved need, failure of others, etc., might be

> utilized to give light to the circumstances surrounding the origin of the subject matter to be patented.

Moreover, in In re Ehrreich and Avery, 200 USPQ 504, 509-510 (CCPA 1979), the Court of Customs and Patent Appeals further clarified the basic test set forth in Graham v. John Deere:

> We must not here consider a reference in a vacuum, but against the background of the other references of record which may disprove theories and speculations in the reference or reveal previously undiscovered or unappreciated problems. The question in a §103 case is what the references would collectively suggest to one of ordinary skill in the art. In re Simon, 461 F.2d 1387, 174 USPQ 114 (CCPA 1972). It is only by proceeding in this manner that we may fairly determine the scope and content of the prior art according to the mandate of Graham v. Deere Company, 383 US 1, 17, 148 USPQ 459, 467 (1966). (Emphasis in original.)

Thus, the mere fact that parts of prior art disclosures can be combined does not make the combination obvious unless the prior art also contains something to suggest the desirability of the combination. In re Imperato, 486 F.2d 585 (CCPA 1973).

To imbue one of ordinary skill in the art with knowledge of the invention, when no prior art reference or references of record convey or suggest that knowledge, is to fall victim to the insidious effect of hindsight syndrome wherein that which only the inventor taught is used against its teacher. W.L. Gore & Assoc. v. Garlock, Inc., 721 F.2d 1540, 1533, 220 USPQ 303, 312-13 (Fed. Cir. 1983).

Further, analyzing the claimed invention as a whole in view of the prior art as a whole, one indicium of nonobviousness is a "teaching away" from the claimed invention by the prior art at the time the invention was made. See U.S. v. Adams, 148 USPQ 479 (1966). Essentially, teaching away from a claimed invention is a per se demonstration of lack of prima facie obviousness.

Where the prior art provides "only general guidance and is not specific as to the particular form of the invention or how to achieve it, [such a suggestion] may make an

approach 'obvious to try,' but it does not make the invention obvious." Ex parte Obukowicz, 27 USPQ2d, 1063, 1065 (U.S. Patent and Trademark Office Board of Appeals and Interferences, 1992) and In re O'Farrell, 7 USPQ2d 1673, 1681 (Fed. Cir. 1988).

Further, in In re Wright, 848 F.2d 1216, 6 USPQ2d 1959 (Fed. Cir. 1988), the Federal Circuit stated:

> Factors including unexpected results, new features, solution of a different problem, novel properties are all considerations in the determination of obviousness...

These secondary considerations (objective evidence of non-obviousness) as outlined in Graham v. John Deere and further characterized in In re Wright must be evaluated before reaching an ultimate decision under 35 U.S.C. §103.

The test for obviousness is what the combined teachings would have suggested to one of ordinary skill in the art. See, In re Young, 927 F.2d 588, 591, 18 USPQ2d 1989, 1091 (Fed. Cir. 1991) and In re Keller, 642 F.2d 413, 425, 208 USPQ 871, 881 (CCPA 1981). More specifically, as stated by the court in Keller, 642 F.2d at 425, 208 USPQ at 881, the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary references; nor is it that the claimed invention must be expressly suggested in one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art. Moreover, the artisan is not compelled to blindly adopt every single aspect of the teachings of any one reference without the exercise of independent judgment, see Lear Siegler, Inc. v. Aeroquip Corp., 733 F.2d 881, 889, 221 USPQ 1025, 1032 (Fed. Cir. 1984).

With regard to motivation to combine the references used in the rejection of Appellant's claims, while there must be some teaching, reason, suggestion or motivation to combine existing elements to produce the claimed device, it is not necessary that the cited references or prior art specifically suggest making the combination. See, B.F. Goodrich Co.

v. Aircraft Braking Systems Corp., 72 F.3d 1577, 1583, 37 USPQ2d 1314, 1319 (Fed. Cir 1996) and In re Nilssen, 851 F.2d 1401, 1403, 7 USPQ2d 1500, 1502 (Fed. Cir. 1988). Rather, the test for obviousness is what the combined teachings of the references would have suggested to one of ordinary skill in the art.

The Office Action must provide proper motivation to combine the teaching of different references. The first requirement of proper motivation is that a showing of a suggestion, teaching, or motivation to combine the prior art references is an "essential evidentiary component of an obviousness holding." C.R. Bard, Inc. v. M3 Sys. Inc., 157 F.3d 1340, 1352, 48 USPQ2d 1225, 1232 (Fed. Cir. 1998). This evidence may flow from the prior art references themselves, the knowledge of one of ordinary skill in the art, or, in some cases, from the nature of the problem to be solved. See Pro-Mold & Tool Co. v. Great Lakes Plastics, Inc., 75 F.3d 1568, 1573, 37 USPQ2d 1626, 1630 (Fed. Cir. 1996). However, the suggestion more often comes from the teachings of the pertinent references. See In re Rouffet, 149 F.3d 1350, 1359, 47 USPQ2d 1453, 1459(Fed. Cir. 1998). This showing must be clear and particular, and broad conclusory statements about the teaching of multiple references, standing alone, are not "evidence." See Dembiczak, 175 F.3d at 1000, 50 USPQ2d at 1617.

The Office Action must also demonstrate that modifying one reference in view of another reference is even feasible. Moreover, the case law requires that for motivation to be proper, showing that something is feasible is not enough. Just because something is feasible does not mean that it is desirable or that one of ordinary skill in the art would be motivated to do what is feasible. See Winner International Royalty Corp. v. Wang, 53 USPQ2d 1580 (Fed. Cir. 2000), which points out that motivation to combine references requires a showing not just of feasibility, but also of desirability.

In Tec Air Inc. v. Denso Manufacturing Michigan Inc., 52 USPQ2d 1294 (Fed. Cir. 1999), the Court of Appeals for the Federal Circuit stated that there is no suggestion to combine relevant teachings from different references if a reference teaches away from its combination with another source. The court also stated that a reference may be said to teach away when a person of ordinary skill in the art, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant.

Additionally, in In re Braat, 16 USPQ2d 1812 (Fed. Cir. 1990) (unpublished), the Court of Appeals for the Federal Circuit reversed a decision by the PTO Board of Appeals and Interferences, stating that the reference upon which the obviousness of claim 1 was based taught away from the claimed invention, and that "[O]ne important indicium of non-obviousness is "teaching away" from the claimed invention by the prior art," citing In re Dow Chemical Co., 5 USPQ2d 1529, 1532 (Fed. Cir., 1988).

Moreover, a factual inquiry whether to modify a reference must be based on objective evidence of record, not merely conclusionary statements of the Examiner. See, In re Lee, 277 F.3d 1338, 1343, 61 USPQ2d 1430, 1433 (Fed. Cir. 2002).

B. Claims 1-5, 8 and 13-16 Are Not Obvious In View of Rubinstein '233 and Wilson

The final Office Action rejects claims 1-5, 8 and 13-16 under 35 USC 103(a) as unpatentable over Rubinstein '233 in view of Wilson. This rejection is improper and should be reversed.

Appellants respectfully submit that this rejection is based solely on speculation, which cannot properly serve as the basis for a rejection.

The final Office Action speculates that "obtaining document constraint descriptors based on user signals" is disclosed by the Abstract; however, no indication is given as to which of the more than 100 words in the abstract allegedly constitute such disclosure. The

rejection also refers to Fig. 2, item 250, which is a query pane, which "enables the user to type a query expression or edit a user expression previously constructed." See column 4, lines 49-54. The final Office Action alleges that Rubinstein '233 uses the logically joined expressions to obtain a displayed constraint descriptor set as applied for document searching, citing Fig. 2, item 250 and col. 4, lines 49-56. However, the query pane 250 only provides for user input terms and does not obtain or display a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations. In column 4, lines 49-56, Rubinstein '233 merely discloses that the query pane 250 can be used to type in or edit a query expression. This simply does not disclose obtaining or displaying a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations.

Rubinstein '233 never mentions document constraint descriptors and the final Office Action fails to indicate how Rubinstein '233 allegedly obtains document constraint descriptors. The abstract only indicates that Rubinstein '233 automatically identifies for a user keyword phrases in a plurality of documents. There is no disclosure indication in Rubinstein '233 that document constraint descriptors are generated in the disclosed processes of identifying keyword phrases in a plurality of documents or generating an abstract for the user.

The final Office Action then speculates that Rubinstein '233 receives user signals indicating a set of attribute value-relations that can apply to documents. There is no disclosure in Rubinstein '233 that the user enters attribute values. Rubinstein '233, for example, never discloses any attribute value relationships, and the final Office Action fails to indicate where any such attribute value relationship allegedly exist in Rubinstein '233. The final Office Action mentions Fig. 2, items 206, 208 and 250 in this regard, for example. However, these items merely provide space for attributes, e.g., keywords. No disclosure of

attribute values or a relationship between attributes and their values is disclosed in Rubinstein '233.

The final Office Action also mentions column 3, lines 34-44, which discusses keywords and relevance codes (which are generated by linguistic analyses). This, however, is not receiving user signals indicating a set of attribute value-relations that can apply to documents. Even if the linguistic analyses generated relevance codes were somehow considered to be attribute value relationships, which they are not, they are not user generated, as recited in claims 1 and 13.

The final Office Action also mentions column 4, lines 1-10, which discloses that a user may generate query expressions in which one or more keyword phrases appear as operands (col. 4, lines 1-22). However, how this allegedly constitutes attribute-value relationships is not clear, and it is not explained in the final Office Action.

The final Office Action then discusses "obtaining logical relations via inclusion of keywords into logic panes to produce logically joined expressions" as allegedly being disclosed in Fig. 2, items 242 , 246 and column 4, lines 17-30, and asks Appellants to "compare with claim 1 'using . . . to obtain logical relations equivalent to the attribute-value relations.'"

Appellants respectfully submit that items 242 and 246, which provide for document keywords, and the ANDing of these search terms together (col. 4, lns. 17-30) is in no way obtaining logical relations equivalent to attribute-value relations input by a user.
As noted above, a user does not enter attribute-value relationships in Rubinstein '233. Even if such relationships were entered, which they are not, the relationships obtained by Rubinstein '233 are merely ANDing of search terms together and have nothing to do with obtaining logical relations equivalent to attribute-value relations input by a user. These ANDed terms are merely a Boolean logical combination of keywords. Reference is made, in this regard, to

the statement on page 11 of Appellants' disclosure that a constraint includes logical combinations of constraints. All that Rubinstein '233 is doing in this regard is expanding the attributes into logical combinations of attributes. Rubinstein '233 is not obtaining logical relations equivalent to attribute-value relations input by a user.

The final Office Action admits that Rubinstein '233 does not specifically teach a "sort." To make up for this admitted shortcoming of Rubinstein '233, the final Office Action turns to Wilson, which discloses an automatic, context-organizing query interface. The final Office Action asserts that Wilson teaches selection of arguments, operations and relations, and that a logical operation may be sorting or other operation, citing column 3, lines 45-55.

Appellants respectfully point out that claims 1 and 13 use the term "sort," which is defined in the paragraph bridging pages 19 and 20 of the specification as a unary relation expressing a property of a single entity, in the context of using the user input signals to obtain, without user intervention, logical relations, which comprise at least one of a sort and a feature, which are equivalent to the attribute value relations. The final Office Action does not demonstrate that either Rubinstein '233 or Wilson disclose a sort as recited in claims 1 and 13.

Turning to the "Response to Argument" portion of the final Office Action, the final Office Action interprets "sort" as disclosed by Wilson. Appellants respectfully submit that the specification of Wilson never defines a "sort." The term "sort" and the term "sorting" are both used once in the specification of Wilson, but the specification of Wilson never defines those terms. If "sort" in Wilson has a different meaning that does "sort" in Appellants' disclosure, then that is prima facie evidence that the meaning of sort needs to be clarified so that one of ordinary skill in the art will know which definition is being used by Appellants. Appellants define "sort" in their application, as pointed out above, as an aid to one of ordinary skill in the art.

Moreover, the alleged motivation to combine Rubinstein '233 and Wilson to incorporate sorts for convenient arrangement of related/ranged results is a broad, general statement of the type which does not provide evidence of the desirability of making the proposed modification. A showing of motivation must be clear and particular, and broad conclusory statements about the teaching of multiple references, standing alone, are not "evidence." See In re Dembiczak, 175 F.3d 994 at 1000, 50 USPQ2d 1614 at 1617. Moreover, the Board of Patent Appeals and Interferences has indicated that the mere fact that teachings found in the prior art could be combined as proposed by an Examiner does not make the combination obvious "absent some teaching, suggestion, or incentive supporting the combination." Ex parte Metcalf, 67 USPQ2d 1633 (BPAI 2003) (nonprecedential).

Appellants respectfully submit that the final Office Action has failed to provide a proper teaching, suggestion, or incentive supporting the asserted combination of Rubinstein '233 and Wilson. The final Office Action only provides the hindsight reason "to release the burden of modification from the user," whereas the applied references fail to indicate that such a burden exists or needs to be relieved. This reason is a broad, general statement of the type which does not provide evidence of the desirability of making the proposed modification.

Accordingly, the final Office Action fails to make out a prima facie case of obviousness of the invention as recited in claims 1 and 13.

The final Office Action also applies Wilson to modify Rubinstein '233 to automatically provide whatever Rubinstein '233 provides, citing column 7, lines 12-21 and column 12, lines 34-48 of Wilson, which allegedly disclose automatically making necessary changes and divisions in Boolean groupings within relations, resulting in proper constraints. Actually, Wilson lets a user select the Boolean operators to use in the search query (col. 7, lns. 12-21) or, if the user selects a Boolean in step 74 of Fig. 2 that is different than previous

Booleans within the same group, the processor may create a new group automatically or let the user create the new group (col. 12, lns. 34-48).

Appellants do not understand what automating a particular decision step in a Boolean operation has to do with the overall process of using user signals, without user intervention, to obtain logical relations equivalent to the attribute value relations, and using the logical relations to obtain, without requiring user intervention, a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations. The final Office Action fails to provide any explanation in this regard. In both Rubinstein '233 and Wilson, the user input includes only logical Boolean operations. The proposed combination of references simply does not disclose the inventions recited in claims 1 and 13.

Rubinstein '233 simply does not disclose obtaining document constraint descriptors based on user signals. The system of Rubinstein '233 generates abstracts of documents on its own and presents the abstracts to a user. The user then creates keyword(s), and the system of Rubinstein '233 then identifies a document based on the keyword(s).

Moreover, the user query in Rubinstein '233 uses logical operators such as "AND" and "BUT NOT" only in the sense that they link query terms so as to expand the query. Rubinstein '233 does not disclose obtaining logical relations equivalent to attribute-value relations input by a user, as claimed. An expansion of a query using logical operators merely expands the query and in no way obtains a logical relation that is equivalent to attribute-value relations input by a user. At best, what is obtained in Rubinstein '233 is an expanded query that constitutes the user expressed attribute value relation. Rubinstein '233 does not go any further, for example, to obtain a logical relation that is equivalent to the user expressed attribute value relation, as claimed.

Appellants' invention as recited in claims 1 and 13 differs significantly from Rubinstein '233 taken alone, or in combination with Wilson. As pointed out on pages 4 and 5

of this application, a typical user has difficulty formulating sorts and features that will produce a desired constraint. These constraint production difficulties are alleviated by the claimed invention which provides techniques that obtain document constraint descriptors from documents from user signals. Instead of requiring a user to provide a set of sorts and features, the claimed invention allows a user to simply provide attribute-value relations, and the invention recited in claims 1 and 13 converts the attribute-value relations to logical relations such as sorts and features from which a constraint descriptor can be obtained.

Neither Rubinstein '233 nor Wilson discloses, teaches or suggests the features recited in claims 1 and 13.

With respect to claims 15 and 16, neither Rubinstein '233 nor Wilson discloses a document constraint descriptor or using a document constraint descriptor to solve a set of one or more constraints to obtain a solution. The search query in column 5, lines 54-61 of Rubinstein '233 simply does not constitute a document constraint descriptor or using a document constraint descriptor to solve a set of one or more constraints to obtain a solution, at least for the reasons stated above regarding the rejection of claim 13.

Claims 2-5, 8 and 14-16 are patentable over Rubinstein '233 and Wilson at least for the reasons presented above regarding claims 1 and 13.

With respect to the assertion in the final Office Action, on page 11, that ". . . Wilson is used to teach a query interface encompassing Boolean operators for relating logical operations . . . resulting in proper constraints," Appellants respectfully submit that this statement only reinforces Appellants' argument that the applied art merely expands query terms resulting in expanded constraints. This is a far cry from obtaining logical relations equivalent to attribute-value relations input by a user, as recited in the claims.

C. **Claims 6 and 7 Are Not Obvious In View of Rubinstein '233, Wilson and Karnik**

The final Office Action rejects claims 6 and 7 under 35 USC 103(a) over Rubinstein '233 in view of Wilson as applied against claim 1, and further in view of Karnik. This rejection is improper and should be reversed.

Karnik is not directed to identifying one of a plurality of documents, as are Rubinstein '233 and Wilson. Instead, Karnik is only interested in entering information and/or extracting information from preselected areas in one or more documents.

The final Office Action stretches this reference combination in the sense that the final Office Action completely fails to establish a motivation to combine these references. The alleged reason to combine these three references is to provide Rubinstein '233 with the capability of querying data from inputted IRS forms for statistical purposes. However, in order to modify Rubinstein '233, one would have to disregard the main function of Rubinstein '233's of identifying one of a plurality of documents and identifying the document by generating an abstract by linguistically analyzing the plurality of documents. One of ordinary skill in the art would not have been motivated to fundamentally alter Rubinstein '233 in this manner. Moreover, Rubinstein '233 is not restricted to querying only certain portions of a form, and there would have been no motivation for one of ordinary skill in the art to so restrict Rubinstein '233.

This rejection is a prime example on improper hindsight reconstruction of Appellants' invention based solely on Appellants' disclosure.

With respect to the assertion on page 11 of the final Office Action that Karnik provides Rubinstein '233 the capability of querying data from a form for statistical purposes, Appellants respectfully submit that Rubinstein '233 has no disclosure or suggestion of a desire to query data from a form for statistical purposes, nor does Rubinstein '233 have any

disclosed need to retrieve information from a computer database and place that data at precise positions on a pre-printed form.

Reference is made, in this regard to the Metcalf decision, cited above, which aptly points out that the mere fact that teachings found in the prior art could be combined as proposed by an Examiner does not make the combination obvious "absent some teaching, suggestion, or incentive supporting the combination."

Accordingly, the final Office Action fails to make out a prima facie case of obviousness of the subject matter recited in claims 6 and 7.

D. Claims 9-12 Are Not Obvious In View of Rubinstein '233, Wilson and Rubinstein '897

The final Office Action rejects claims 9-12 under 35 USC 103(a) over Rubinstein '233 in view of Wilson as applied against claim 1, and further in view of Rubinstein '897. This rejection is improper and should be reversed.

The final Office Action only addresses the network feature of claim 9. However, the network feature is not the only positively recited feature of claim 9. Claim 9 also recites solving the set of one or more constraints to obtain a solution and using the solution to obtain one or more document references. The final Office Action does not even address these positively recited features, thereby denying Appellants fundamental procedural and substantive due process under the Administrative procedures Act. For this reason alone, the final Office Action fails to make out a prima facie case of obviousness of the subject matter recited in claim 9.

Moreover, the applied references do not teach these features, at least for the reasons stated above regarding the rejection of claim 1.

Claims 10-12 recite the features of claim 1 and are patentable at least for the reasons stated above regarding the patentability of claim 1 with respect to Rubinstein '233 and Wilson. It is noted that Rubinstein '897 is only applied to teach creating logical relations

using the Internet, and not to remedy any of the aforementioned deficiencies in Rubinstein '233 and Wilson.

VI. REBUTTAL OF ARGUMENTS IN THE ADVISORY ACTION

The Advisory Action contains the Examiner's rebuttal arguments, a number of which are addressed to some extent above. However, in order to fully respond to those arguments, the following is presented.

The Advisory Action states that "[S]ince the query within item 250 of Figure 2 can result from previous construction as explained above (See Rubinstein '233 column 4, lines 49-55), said query can be interpreted as 'document constraint descriptors' (associated with said relevance codes), because it acts as a descriptor to constrain data so as to fit said query."

Appellants respectfully disagree, and submit that the query pane 250 merely shows query expressions formulated by a user. As explained in column 4, lines 49-59, the "[Q]uery pane 250 enables the user to type a query question or edit a query question previously constructed by the drag and drop technique described above." It also "presents a history of prior query expressions that may be recalled queries typed in by a user." (Col. 4, lns. 56-59). User queries in the query pane 250 of Rubinstein merely indicate a set of attribute relations that can apply to documents. They are not "a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations equivalent to the attribute relations," "comprising one of at least a sort and a feature," "obtained without user intervention," as recited in the pending claims.

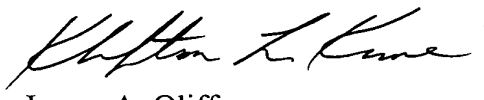
With respect to the meaning of the word "sort", Appellants respectfully submit that the term "sort" is positively defined in the specification and that they are their own lexicographers in this regard.

CONCLUSION

The final Office Action fails to make out a prima facie case of obviousness of the claimed invention for the reasons stated above, and the rejection of claims 1-16 should be reversed.

The Honorable Board is requested to reverse the rejections set forth in the Final Rejection and to pass this application to issuance.

Respectfully submitted,



James A. Oliff
Registration No. 27,075

Klifton L. Kime
Registration No. 42,733

JAO:KLK/kzb

Attachment:
Appendix

Date: February 2, 2004

OLIFF & BERRIDGE, PLC
P.O. Box 19928
Alexandria, Virginia 22320
Telephone: (703) 836-6400

DEPOSIT ACCOUNT USE AUTHORIZATION
Please grant any extension necessary for entry;
Charge any fee due to our Deposit Account No. 15-0461

APPENDIX A

CLAIMS:

1. A method for obtaining document constraint descriptors based on user signals, the method comprising:

(A) receiving user signals indicating a set of attribute-value relations that can apply to documents;

(B) using the user signals to obtain, without requiring user intervention, logical relations equivalent to the attribute value relations, the logical relations comprising at least one of a sort and a feature; and

(C) using the logical relations to obtain, without requiring user intervention, a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations.

2. The method of claim 1 in which the method is performed with a machine that includes user interface circuitry for receiving user signals and in which (A) comprises:

(A1) receiving a series of one or more user signals through the user interface circuitry, the series of user signals indicating the attribute-value relations.

3. The method of claim 2 in which the series of user signals includes two or more user signals.

4. The method of claim 2 in which the machine is a portable computing device and the user interface circuitry includes a touchscreen or a keyboard.

5. The method of claim 2 in which the machine is a fixed computing device and the user interface circuitry includes one or more of a touchscreen, a keyboard, and a mouse.

6. The method of claim 2 in which the machine is a multifunction device, in which the user interface circuitry includes a scanner, and in which (A1) comprises:

scanning at least a part of an image-bearing portable medium to produce electronic signals; and

using the electronic signals to obtain the series of one or more user signals.

7. The method of claim 6 in which the medium is a form that includes one or more fields, at least one of the fields having a human readable indication of an attribute and an area that a user can mark to indicate a set of at least one value of the attribute.

8. The method of claim 2 in which the user interface circuitry includes display circuitry for presenting images to a user and selection circuitry the user can operate to provide signals indicating items in images presented by the display circuitry, in which the machine includes memory, and in which (C) comprises:

storing the document constraint descriptor in memory;

the method further comprising:

(D) presenting an image through the display circuitry that includes an item representing the document constraint descriptor;

(E) receiving a user signal from the selection circuitry indicating the item; and

(F) in response to the user signal, obtaining the stored document constraint descriptor.

9. The method of claim 1 in which the machine is connected through a network for accessing a repository of electronic documents; the method further comprising:

(G) solving the set of one or more constraints to obtain a solution and using the solution to obtain one or more document references, each document reference indicating an electronic document in the repository that satisfies the set of one or more constraints.

10. The method of claim 9 in which the user interface circuitry includes display circuitry for presenting images to the user and selection circuitry the user can operate to provide signals indicating items in images presented by the display circuitry; the method further comprising:

(H) presenting an image through the display circuitry that includes, for each

document reference, an item representing the document reference;

(I) receiving a user signal through the selection circuitry indicating a first item representing one of the document references; and

(J) in response to the user signal, retrieving from the repository the electronic document indicated by the first item's document reference.

11. The method of claim 10 in which (J) comprises:

presenting a portion of the electronic document through the display circuitry.

12. The method of claim 10 in which the machine further includes printing circuitry and in which (J) comprises:

operating the printing circuitry to print the electronic document.

13. A machine for obtaining document constraint descriptors based on user signals, the machine comprising:

a processor; and

user interface circuitry for providing user signals to the processor;

the processor operating to:

receive user signals through the user interface circuitry indicating a set of attribute-value relations that can apply to documents;

use the user signals to obtain, without requiring user intervention, logical relations equivalent to the attribute-value relations, the logical relations comprising at least one of a sort and a feature; and

use the logical relations to obtain, without requiring user intervention, a document constraint descriptor defining a set of one or more constraints equivalent to the logical relations.

14. The machine of claim 13 in which the machine is a portable computing device.

15. The machine of claim 13 in which the machine is a fixed computing device and the machine is connected to a repository of electronic documents that includes one or more documents that satisfy the set of one or more constraints, the processor further operating to:

use the document constraint descriptor to solve the set of one or more constraints to obtain a solution, and use the solution to obtain one or more document references, each document reference indicating a document that satisfies the set of one or more constraints.

16. The machine of claim 15 in which the user interface circuitry includes display circuitry for presenting images to a user and selection circuitry the user can operate to provide signals indicating items in images presented by the display circuitry; the processor further operating to:

present an image through the display circuitry that includes, for each document reference, an item representing the document reference;

receive a user signal through the selection circuitry indicating an item representing a selected one of the document references; and

in response to the user signal, access the selected document reference.